REMARKS

The present response is to the Office Action mailed in the above-referenced case on August 06, 2007. Claims 1-35 are standing for examination. The Examiner rejects claims 1-11 under 35 U.S.C. 101. Claims 1-35 are rejected under 35 U.S.C. 103(a) as being unpatentable over Simons et al. (US 6,332,198) hereinafter Simons, in view of Zadikian et al. (US 6,724,757) hereinafter Zadikian.

In response, applicant herein amends claim 1 to change the word "distribution" to "distributed" in order to overcome the 101 rejection. Applicant provides further arguments to overcome the rejections of the claims on their merits. Applicant's arguments will clearly distinguish applicant's patentable subject matter over the combined teachings of Simons and Zadikan.

Applicant reproduces and responds to the Examiner's "Response to Argument" portion, kindly presented in the present Office Action:

The Examiner states; "Applicant's arguments with respect to claims 1-35 have been considered but are moot in view of the new ground(s) of rejection."

Applicant argues that applicant's arguments are in fact not moot because there is no new merit rejection from the Examiner that affects applicant's arguments as the Examiner has maintained the same art of Simons and Zadikian and as evidenced by the Examiner's response in the present Office Action to said arguments.

The Examiner states; "Applicant argues (Remarks, page 9) that the "software backup" of Simons is not APS software, as claimed, Further, there is absolutely no evidence in the art of Simons that APS software backup is implemented by direct communication between the involved software modules (primary & backup) at the interfaces. Examiner respectfully disagrees. Simons clearly discloses: modular software architecture, software intelligence is stored locally. Furthermore, Simons discloses that redundancy designs come in two forms: 1:1 and 1:N. In a so-called 1:1 redundancy design a backup element exists for every active or primary element. To minimize synchronization time, many 1:1 redundancy schemes support hot backup of software,

which means that the software on the backup elements mirror the software on the primary elements (column 1, line 46-column 2, line 31; column 45, lines 61-67)."

Applicant respectfully points out that applicant argued in full context that the "software backup" of Simons is not APS software, as claimed. Further, there is absolutely no evidence in the art of Simons that APS software backup is implemented by direct communication between the involved software modules (primary & backup) at the interfaces. Simons teaches a hierarchical fault management system which clearly discloses that; "When master SRM 36 (on master processor) detects or receives notice of a failure or event, it notifies slave logging entity 433a, which notifies master logging entity 431. Master SRM 36 also determines the appropriate corrective action based on the type of failure or event its fault policy. Corrective action may require failing-over one or more line cards 16a-16n or other boards, including central processor 12, to redundant backup boards (col. 35, line 58 to col. 36, line 3). In every instance of backup in the art of Simons, a master application directs the switch-over between the primary and backup devices.

Applicant believes that Examiner should consider the functions of Simons' Master SRM 36, as argued. The Examiner's ignorance of the teaching of Simons as presented by applicant above is not due diligence in the examination process. Applicant proves that the "software backup" as taught in Simons is not APS software because Simons specifically teaches how the Master SRM 36 deals with detected failures at line cards.

The Examiner points out the teaching in the background portion of Simons describing backup processes as known in the art wherein 1:1 and 1:N synchronization schemes occur. This teaching refers to a system where there is one designated backup device to one primary device, or one backup device for many primary devices.

Applicant argues that the present invention, as claimed, teaches an N:N redundancy scheme because APS switchover is solely accomplished through direct communication between the APS client modules running on the processors supporting the involved interfaces. Applicant points out the benefit of applicant's invention because in Simons' teaching if the designated backup fails also, the network would fail. In

applicant's invention all devices serve as both primary and backup devices for each other, at all times, which is unknown in the art at the time of filing applicant's invention.

The Examiner states; "Still on page 10, Applicnt argues that "in Simons, information and communication needed to facilitate true APS is not stored locally in software of each individual APS module, as in applicant's invention and claims, the 50 millisecond time frames could not be accomplished as claimed." Examiner respectfully disagrees. As shown in Figs 1, 5, and 33, computer system 10 includes multiple line cards 16a-16n. Each line card includes a control processor subsystem 18a- 18n, which runs an instance of the kernel 22a-22n including slave and client programs as well as line card specific software applications. Each control processor subsystem 14, 18a-18n operates in an autonomous fashion. This shows that software is adapted to run on multiple-processor. Furthermore, Simons clearly discloses a distributed redundancy architecture that spreads software backup (hot state) across multiple elements (column 39, lines 43-48; line 62-column 40, line 12). In addition, Simons discloses that modular software architecture dynamically loads applications as needed by gathering necessary information (i.e., metadata) from a variety of sources. Metadata provides seamless extensibility allowing new software processes to be added and existing software processes to be upgraded or downgraded while the operating system is running (column 6, line 55-column 7, line 12). This shows that true APS is accomplished without data flow interruption.

Applicant respectfully disagrees with the Examiner's above statement; "This shows that true APS is accomplished without data flow interruption". This is clearly a case of the Examiner adding subject matter to the teaching of Simons which is not clearly taught in Simons. As previously argued, the software the Examiner refers to in Simons is not software to perform APS, as claimed. Simons does not teach that it is software to perform APS. Simon's clearly teaches how faults are handled using the Master SRM 36, as argued above.

The Examiner states; "Applicant argues that "Zadikian does not teach or suggest 50ms switchover in a distributed processing system, as taught in applicant's invention, as claimed". It is respectfully submitted that the rejection is based on the combined teaching of the Simons reference and the Zadikian reference and that the Simons reference, as

pointed out above does teach a distributed processing system. Furthermore, Zadikian teaches that router 1 00 supports a restoration of a majority of network failures within less than 50m."

A strongly argues that the fact that Zadikian's scheme helps ensure (not consistently achieves or supports) 50-millisecond switchover in a single processor implementation certainly would not obviate a 50-millisecond switchover in a distributed processor environment as taught in Simons. Simply because the Examiner has produced art teaching that a 50-ms switchover exists (in questionable analogous art) does not teach or suggest 50ms switchover in a distributed processing system, as taught in applicant's invention, as claimed. Applicant believes the two pieces of art need a technical or physical capacity to work together to accomplish applicant's invention, as claimed. Applicant argues that because both Simons and Zadikian are not capable of working in a distributed fashion pulling all data required for the APS switchover from a local source, the combination would fail.

Applicant points out that the independent claims of applicant's invention specifically recite; "*APS client modules running on second ones of the multiple processors, the APS client modules for monitoring interface state information, reporting to the APS server application, and for negotiating with other APS client modules;*

characterized in that all of the APS software-dependent data resides locally in APS software of each individual APS module, and further characterized in that APS interface relocation from a primary interface to a backup interface is performed, via said software, through direct communication between the APS client modules..."

Applicant again argues that the Examiner has not produced valid art teaching the limitations of applicant's claims as presented above and argued extensively. In Simons, information and communication needed to facilitate true APS is not stored locally in software of each individual APS module, as in applicant's invention and the 50 millisecond time frames could not be accomplished as claimed. Applicant therefore strongly maintains that Simon suffers from network data flow interruption because true APS is not accomplished.

Applicant therefore believes that claims 1, 12 and 24 as argued by applicant are clearly and unarguably patentable over the art of Simon and Zadikian, either singly or combined. Claims 2-11, 13-23, and 25-35 are then patentable on their own merits, or at least as depended from a patentable claim.

As all of the claims as amended and argued are clearly shown to be patentable over the prior art, applicant respectfully requests that the rejections be withdrawn and that the case be passed quickly to issue. If any fees are due beyond fees paid with this response, authorization is made to deduct those fees from deposit account 50-0534. If any time extension is needed beyond any extension requested with this amendment, such extension is hereby requested.

Respectfully Submitted,
Sundara Murugan

By /Donald R. Boys/
Donald R. Boys
Reg. No. 35,074

Central Coast Patent Agency, Inc.
3 Hangar Way, Suite D
Watsonville, CA 95076
831-768-1755